

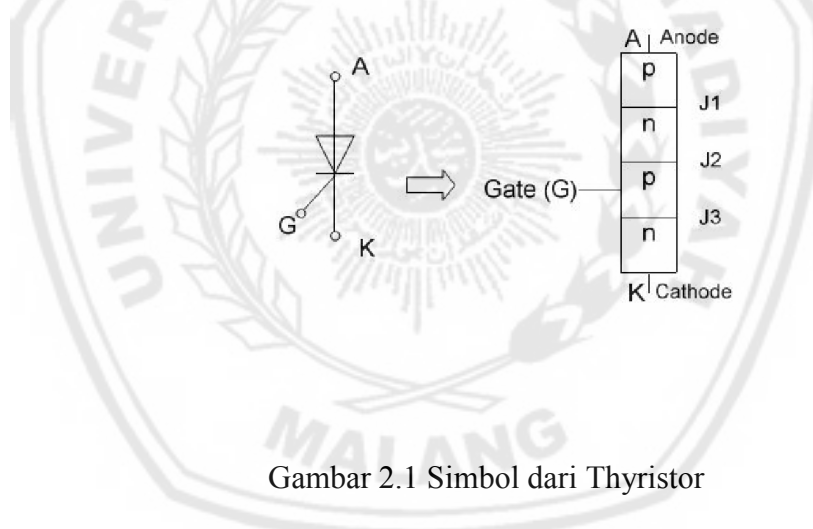
BAB II

Tinjauan Pustaka

2.1 Thyristor

Thyristor merupakan peralatan elektronik yang terdiri dari empat lapisan semikonduktor pnpn dan memiliki tiga sambungan-pn. Thyristor memiliki 3 terminal, yaitu anoda, katoda, dan *gate* (gerbang), seperti ditunjukkan pada gambar

1. Thyristor disebut juga dengan penyearah terkendali, karena memiliki *gate* yang berfungsi untuk mengendalikan arus. Teknik penyalan thyristor yang sering di arus pada gerbang dengan memakai tegangan positif antara gerbang dan terminal katoda akan menyalakan thyristor. Penundaan pemberian arus pada gerbang thyristor tergantung pada operasi *zero crossing*. Thyristor dalam keadaan menyala dapat dipadamkan dengan mengurangi arus maju sampai di bawah arus holding (IH).

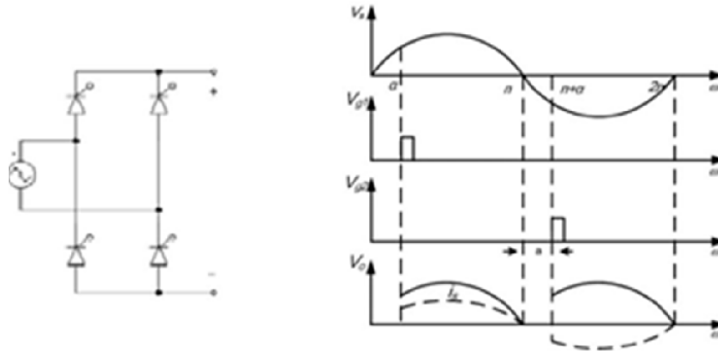


Gambar 2.1 Simbol dari Thyristor

Teknik komutasi adalah salah satu teknik pemadaman thyristor. Dalam teknik komutasi, arus anoda dijaga agar berada dibawah arus holding dalam waktu yang cukup lama. Jadi semua pembawa muatan lebih pada 4 lapisan pnpn dapat dihapuskan.

2.2 Penyearah Gelombang Penuh Terkontrol Satu Fasa.

Rangkaian penyearah terbagi atas beberapa macam. Gambar 1 adalah rangkaian dan bentuk gelombang tegangan dan sinyal *trigger* untuk rangkaian penyearah terkendali satu fasa gelombang penuh.



Gambar 2.2 Bentuk Rangkaian dan Gelombang Keluaran dari Penyearah Terkontrol Satu Fasa Gelombang Penuh.

Selama setengah siklus positif thyristor T_1 dan T_2 dalam keadaan bias maju, dan diberi sudut penyalan sebesar $\omega t = \alpha$, Selama setengah siklus negatif T_3 dan T_4 akan dibias maju, dan penyalan dari thyristor T_3 dan T_4 menggunakan tegangan yang mengalir melalui thyristor T_1 dan T_2 sebagai tegangan *reverse blocking*. Tegangan V_{dc} yang dihasilkan adalah:

$$V_{dc} = \frac{2}{2\pi} \int_{\alpha}^{\pi} V_m \sin \omega t \, d(\omega t) = \frac{V_m}{\pi} (1 + \cos \alpha) \dots\dots\dots (2.1)$$

2.3 Algoritma Particle Swarm Optimization (PSO)

Particle swarm optimization (PSO) diperkenalkan pada tahun 1995 oleh Kennedy dan Eberhart pada tahun 1995 [1, 2]. PSO adalah suatu metode pencarian nilai optimal yang terinspirasi oleh kegiatan/pola dari burung dan ikan dalam mencari makanan. Sifat burung dan ikan yang mencari makanan dalam suatu kelompok besar sehingga jika salah satu anggota menemukan makanan maka akan memberitahukan kepada yang lain agar semua anggota dapat ikut menikmati, menunjukkan suatu pola optimisasi yang jelas. Yaitu bagaimana mencapai tujuan dengan cepat dan tepat.

Suatu teknik optimisasi yang berbasis pada populasi. Populasi PSO disebut dengan *swarm*. Pada sistem PSO beberapa kandidat yang telah diinisialisasi kemudian dievaluasi secara bersamaan. Setiap kandidat yang menghasilkan solusi disebut sebagai partikel. Karakteristik dari partikel ini adalah akan bergerak pada ruang lingkup permasalahan untuk mencari nilai optimum. Sebuah partikel yang bergerak akan belajar dari pengalaman partikel tersebut yang terdahulu seiring dengan Bergeraknya waktu. Setiap partikel akan

selalu mengatur posisi diri sendiri berdasarkan pada pengalaman dari partikel di sekitar. Apabila suatu partikel menemukan posisi yang terbaik untuk menghasilkan nilai optimal, maka partikel-partikel lain di sekitarnya akan bergerak untuk mendekati posisi tersebut. Partikel-partikel tersebut akan bergerak dengan kecepatan yang berbeda-beda untuk mendekati posisi terbaik yang ditemukan oleh partikel terdahulu. Kecepatan suatu partikel untuk bergerak dapat dihitung dengan menggunakan informasi dari:

- (i) kecepatan sekarang,
- (ii) jarak antara posisi semula dengan posisi terbaik yang telah ditemukan.

Sebagai contoh, misalnya perilaku burung-burung dalam kawanan burung. Meskipun setiap burung mempunyai keterbatasan dalam hal kecerdasan, biasanya ia akan mengikuti kebiasaan seperti berikut :

- a. Selalu dekat dengan burung yang lain.
- b. Burung tersebut akan mengarahkan terbangnya ke arah rata-rata keseluruhan burung.
- c. Akan memposisikan diri dengan rata-rata posisi burung yang lain dengan menjaga sehingga jarak antar burung dalam kawanan itu tidak terlalu jauh.

Dengan demikian perilaku kawanan burung akan didasarkan pada kombinasi dari 3 faktor simpel berikut:

- 1. Kohesi - terbang bersama
- 2. Separasi - jangan terlalu dekat
- 3. Penyesuaian - mengikuti arah bersama

Jadi PSO dikembangkan dengan berdasarkan pada model berikut:

- a. Ketika seekor burung mendekati target atau makanan (bisa minimum atau maximum suatu fungsi tujuan) secara cepat mengirim informasi kepada burung-burung yang lain dalam kawanan tertentu.
- b. Burung yang lain akan mengikuti arah menuju ke makanan tetapi tidak secara langsung.

- c. Ada komponen yang tergantung pada pikiran setiap burung, yaitu memorinya tentang apa yang sudah dilewati pada waktu sebelumnya.

Model ini akan disimulasikan dalam ruang dengan dimensi tertentu dengan sejumlah iterasi sehingga di setiap iterasi, posisi partikel akan semakin mengarah ke target yang dituju (minimasi atau maksimasi fungsi). Ini dilakukan hingga maksimum iterasi dicapai atau bisa juga digunakan kriteria penghentian yang lain. Meskipun beberapa modifikasi pada algoritma swarm asli telah dilakukan untuk meningkatkan kinerja dan disesuaikan dengan jenis masalah tertentu, versi seri sebelumnya telah sukses diimplementasikan. Berdasarkan konsep/ analogi dari PSO diatas, maka algoritma PSO secara lebih jelas dapat dituliskan secara matematis sebagai berikut:

- a. Setiap partikel individu i memiliki sifat-sifat sebagai berikut: Posisi saat ini di ruang pencarian, x_{id} , dengan kecepatan gerakan, v_{id} dan sebuah posisi perseorangan terbaik di sebuah ruang pencarian p_{id} .
- b. Sebuah posisi perseorangan terbaik, p_{id} , sesuai dengan posisi dalam ruang pencarian di mana partikel i menghadirkan sebuah kesalahan terkecil yang ditentukan oleh fungsi obyektif f , dengan asumsi memiliki tugas untuk meminimalisasi kesalahan tersebut.
- c. Posisi terbaik global yang ditandai oleh kehadiran posisi partikel yang menghasilkan kesalahan terkecil di antara semua partikel tersebut.

Selama iterasi setiap partikel dalam swarm diperbarui dengan menggunakan dua persamaan berikut:

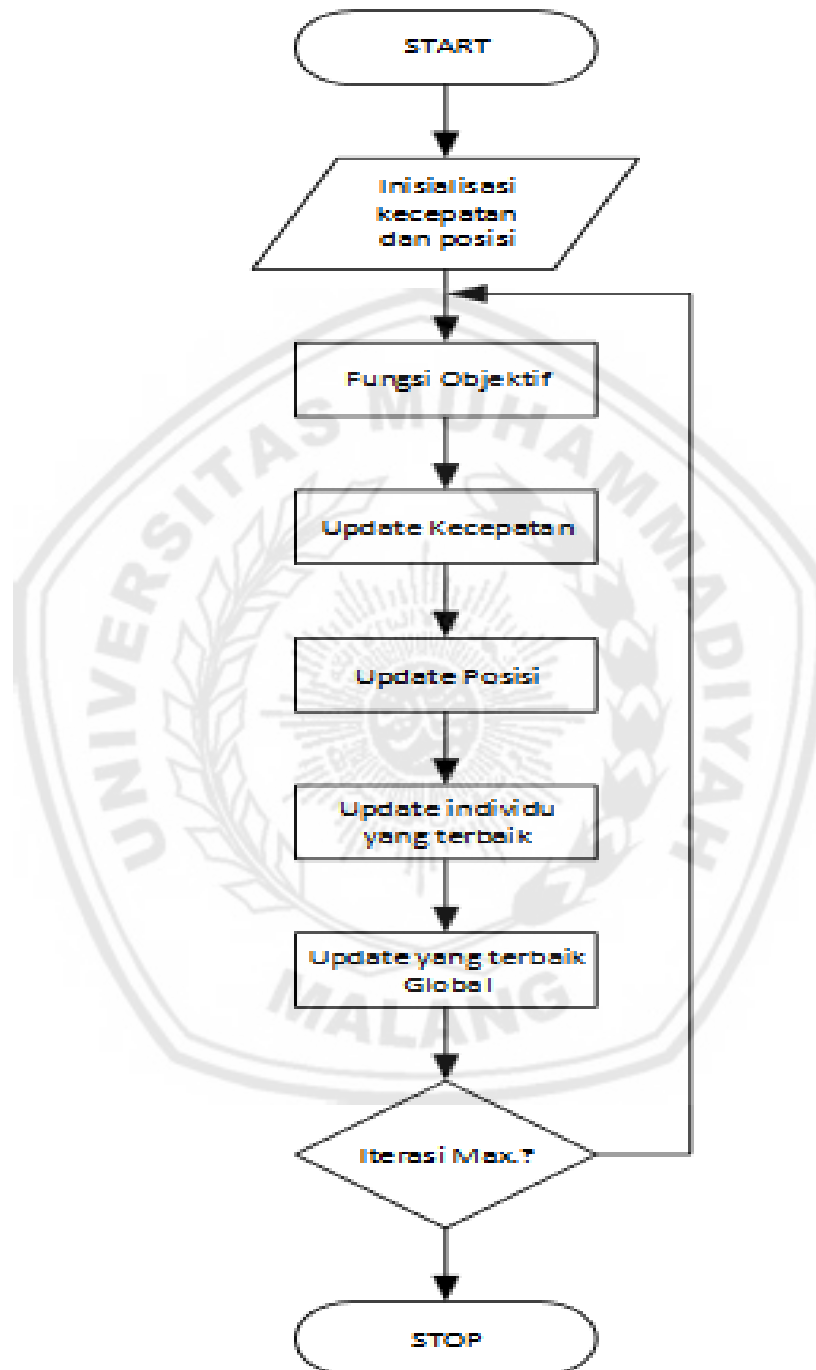
$$v_{id}(t+1) = w \cdot v_{id}(t) + c_1 r_1 (p_{id} - x_{id}(t)) + c_2 r_2 (p_{gd} - x_{id}(t)) \dots \dots \dots (2.2)$$

$$x_{id}(t+1) = x_{id}(t) + v_{id}(t) \dots \dots \dots (2.3)$$

Dimana $v_{id}(t+1)$ dan $v_{id}(t)$ merupakan kecepatan terbaru dan kecepatan saat ini, $x_{id}(t+1)$ dan $x_{id}(t)$ merupakan posisi saat ini. c_1 dan c_2 adalah konstanta positif, dan r_1 dan r_2 merupakan angka acak yang berada pada matriks $[0,1]$ dan w merupakan momen berat.

Secara umum diagram alur dari PSO ditampilkan pada gambar 2.3 sebagai berikut, dimana proses dimulai dengan inisialisasi letak posisi dan kecepatan partikel. Kemudian menentukan kecepatan dari masing-masing partikel hingga ditemukan posisi dari yang

dicari. Proses ini akan terus berlangsung hingga didapatkan posisi terbaik. Proses pencarian nilai optimal akan berhenti setelah iterasi maksimum tercapai, sebelum iterasi itu tercapai maka proses pencarian posisi baru dan proses perkembangan kecepatan akan terus terjadi.



Gambar 2.3 Diagram alur dari algoritma PSO

2.4 Kriteria Optimasi

Sebuah masalah umum dalam desain sistem kontrol adalah menetapkan nilai (konstanta) pengontrol yang sesuai. Secara umum sebuah nilai (konstanta) yang rendah dapat menghasilkan satu respon sistem lambat, sementara dengan nilai yang tinggi dapat menyebabkan satu respon terlalu sering berosilasi dengan kemungkinan ketidakstabilan. Di antara keadaan respon yang ekstrim ini, maka di perlukan satu nilai (konstanta) yang menghasilkan respon sistem terbaik.

Fungsi penting dari umpan balik suatu sistem kontrol harus mengurangi nilai *error* ($e(t)$) antara beberapa variabel sehingga menuntunnya ke nilai nol secepat mungkin. Oleh karena itu, ukuran apapun digunakan untuk ukuran kualitas respon sistem harus mempertimbangkan variasi dari nilai *error* pada rentan waktu keseluruhan. Empat ukuran-ukuran dasar adalah biasa digunakan yaitu:

$$\text{Integral of absolute error (IAE)} = \int_0^{\infty} |e(t)|.dt$$

$$\text{Integral of time multiplied by absolute error (ITAE)} = \int_0^{\infty} t. |e(t)|.dt$$

$$\text{Integral of squared error (ISE)} = \int_0^{\infty} \{e(t)\}^2. dt$$

$$\text{Integral of time multiplied by squared error (ITSE)} = \int_0^{\infty} t\{e(t)\}^2. dt$$

Untuk beberapa kriteria yang mungkin, respon terbaik disesuaikan dengan nilai minimum dari kriteria di atas yang dipilih. IAE sering kali digunakan pada simulasi digital dari sebuah system, tetapi itu tidak sesuai untuk pekerjaan yang analisis, karena nilai mutlak dari suatu fungsi *error* bukan secara umum analitik bentuknya. Masalah ini dapat di atasi dengan menggunakan criteria ISE. ITAE dan ITSE mempunyai tambahan satu pengali waktu dari fungsi *error*, yang menunjukan jangka waktu (*duration*) dari *error* tersebut, dan oleh karena itu, cara ini sebagian besar sering kali diterapkan dalam sistem yang memerlukan waktu penyelesaian cepat.

2.5 Sistem Kontrol PID

Kontrol PID merupakan kombinasi dari tiga jenis kontroler. Kontroler PID juga di masukan dalam sistem kontroler pada saat di inginkan perbaikan tanggapan peralihan maupun tanggapan dalam keadaan mantap. Fugsi alih PID di tulis

$$G(s) = K_d + K_{ds} + K_{1/s} \dots\dots\dots (2.3)$$

Kontroler PID memiliki dua *zero* dan satu *pole*. Salah satu *metode* yang digunakan untuk merancang kontroler PID adalah merancang bagian *Proporsional Integral* (PI) untuk memberikan tanggapan keadaan mantap yang memuaskan selanjutnya kontroler PI ini dianggap sebagai bagian dari proses, *plant* dari bagian *Proporsional Derivative*(PD) dirancang untuk memperbaiki tanggapan peralihan. Adapun persamaan pengontrolan PID adalah

$$mv(t) = K_p(e(t) + \frac{1}{T_i} \int_0^t e(t) dt + T_d \frac{de(t)}{dt}) \dots\dots\dots (2.4)$$

Keterangan :

$mv(t)$ = output dari pengontrol PID atau *Manipulated Variable*

K_p = konstanta *proporsional*

T_i = konstanta *integral*

T_d = konstanta *derivatif*

$e(t)$ = *error*

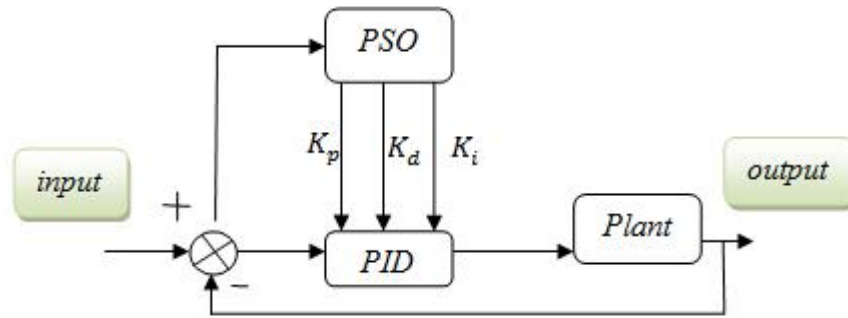
Persamaan Pengontrol *PID* diatas dapat juga dituliskan sebagai berikut:

$$mv(t) = K_p e(t) + K_i \int_0^t e(t) dt + K_d \frac{de(t)}{dt} \dots\dots\dots (2.5)$$

dengan :

$$K_p \times \frac{1}{T_i} \text{ dan } K_d = K_p \times T_d \dots\dots\dots (2.6)$$

Kontroler PSO - PID untuk *plant* ditunjukkan pada gambar dibawah ini :



Gambar 2.4 Struktur Kontroler PSO - PID

Di mana *indeks* performa ITAE yang dipakai untuk mengestimasi parameter-parameter PID diberikan sebagai berikut:

$$ITAE = \int_0^T t |e(t)| dt \dots\dots\dots(2.7)$$

Konsep utama *tuning* kontroler PID secara *on line* adalah *tuning* parameter PID tiap *sampling* waktu. Fungsi obyektif atau *fitness function* yang akan dioptimasi dinyatakan sebagai berikut :

$$J(i) = \alpha ITAE + \beta |o(t)| \dots\dots\dots(2.8)$$

Dimana :

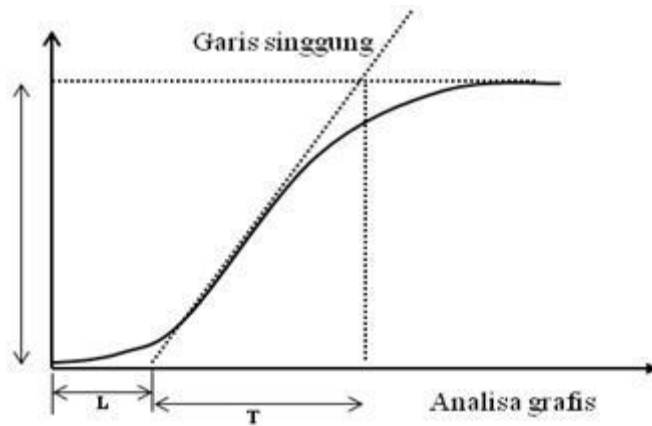
α, β : *factor improment*

o : *overshoot*

ITAE = Integral Time-weighted Absolut Error

2.6 Peranan Nichols-Ziegler

Ziegler – Nichols mengusulkan aturan untuk menentukan nilai K_p , T_i dan T_d berdasarkan pada karakteristik tanggapan peralihan dari plant yang diberikan. Metode pertama Ziegler – Nichols menentukan nilai K_p , T_i , dan T_d :



Gambar 2.5 Analisa Grafis *Ziegler Nichols*

Aturan perpotongan garis lurus terjadi pada kondisi linier dari kurva S repon sistem. Ketepatan dalam pengambilan perpotongan ini sangatlah penting karena menentukan parameter T dan L yang menjadi acuan dari kontroler.

Tabel 2.1 Formula Ziegler Nichols

Tipe Pengendali	K_p	T_i	T_d
P	$\frac{T}{L}$	∞	0
PI	$0.9 \frac{T}{L}$	$\frac{L}{0.3}$	0
PID	$1.2 \frac{T}{L}$	$2L$	$0.5L$